

[> home](#) [> about](#) [> feedback](#) [> login](#)

US Patent &amp; Trademark Office

## Search Results

Search Results for: [redundancy exlining]

Found 3 of 111,041 searched. → Rerun within the Portal

Search within Results

[> Advanced Search](#) [> Search Help/Tips](#)




---

**Sort by:** [Title](#) [Publication](#) [Publication Date](#) [Score](#)  [Binder](#)

---

**Results 1 - 3 of 3**   **short listing**

---

- 1** Procedure exlining: a new system-level specification 87%  
 transformation  
Frank Vahid  
Proceedings of European design automation conference with  
EURO-VHDL '95 on EURO-DAC '95 with EURO-VHDL '95 December  
1995
- 2** Partitioning sequential programs for CAD using a three-step 84%  
 approach  
Frank Vahid  
ACM Transactions on Design Automation of Electronic Systems  
(TODAES) July 2002  
Volume 7 Issue 3  
Many computer-aided design problems involve solutions that  
require the partitioning of a large sequential program written in a  
language such as C or VHDL. Such partitioning can improve design  
metrics such as performance, power, energy, size, input/output  
lines, and even CAD tool run-time and memory requirements, by  
partitioning among hardware modules, hardware and software  
processors, or even among time-slices in reconfigurable computing  
devices. Previous partitioning approaches typically presel ...
- 3** A three-step approach to the functional partitioning of large 77%  
 behavioral processes

[> home](#) [> about](#) [> feedback](#) [> login](#)

US Patent &amp; Trademark Office

## Search Results

Search Results for: [intermediate code<AND>((peephole optimization) )]  
Found 61 of 111,041 searched. → Rerun within the Portal

Search within Results

[> Advanced Search](#) [> Search Help/Tips](#)

Sort by: **Title** **Publication** **Publication Date** **Score** **Binder**

Results 1 - 20 of 61 short listing

Prev  
Page

1

2

3

4

Next  
Page

- |          |   |     |
|----------|---|-----|
| <b>1</b> | Using Peephole Optimization on Intermediate Code  | 95% |
|          | Andrew S. Tanenbaum , Hans van Staveren , Johan W. Stevenson<br>ACM Transactions on Programming Languages and Systems (TOPLAS)<br>January 1982<br>Volume 4 Issue 1  |     |
| <b>2</b> | A language for writing code generators  | 91% |
|          | C. W. Fraser<br>ACM SIGPLAN Notices , Proceedings of the SIGPLAN '89 Conference<br>on Programming language design and implementation June 1989<br>Volume 24 Issue 7 |     |
| <b>3</b> | Code selection through object code optimization   | 87% |
|          | Jack W. Davidson , Christopher W. Fraser<br>ACM Transactions on Programming Languages and Systems (TOPLAS)<br>October 1984<br>Volume 6 Issue 4                      |     |
| <b>4</b> | Fast code generation using automatically-generated decision   | 82% |